ESD-TR-91-243

(2)

Annual Report

Knowledge-Based System Analysis and Control

30 September 1991

Lincoln Laboratory

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

LEXINGTON, MASSACHUSETTS

Prepared for the Department of the Air Force under Contract F19628-90-C-0002.

This report is based on studies performed at Lincoln Laboratory, a center for research operated by Massachusetts Institute of Technology. The work was sponsored by the Department of the Air Force under Contract F19628-90-C-0002.

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

Hugh L. Southall

Hugh L. Southall, Lt. Col., USAF
Chief, ESD Lincoln Laboratory Project Office

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
LINCOLN LABORATORY

KNOWLEDGE-BASED SYSTEM ANALYSIS AND CONTROL

ANNUAL REPORT SUBMITTED TO
CAPT. TONY HATTEN
RL/C3D4
GRIFFISS AFB, NY 13441

H.M. HEGGESTAD
Group 21

1 OCTOBER 1990 – 30 SEPTEMBER 1991

24 JANUARY 1992

Approved for public release; distribution is unlimited.

LEXINGTON MASSACHUSETTS

Abstract

A major aspect of work in FY91 has been the development of MITEC systems for field deployment. A decision was made by Rome Laboratory and the Air Force Communication Command to undertake the development of an Early Release version of MITEC for delivery in FY92 as well as the MITEC (Release 1.0) system started in FY90 and scheduled for delivery in early FY93. The development plan involves the cooperative effort of three Air Force organizations, Rome Laboratory (RL/C3DA), the Air Force Communications Command Communications Systems Center (AFCC/CSC) and Technical Integration Center (AFCC/TIC), with Lincoln Laboratory and its subcontractor Structured Systems and Software, Inc. (3S). The decision to develop MITEC (Early Release), also referred to as MER, in parallel with MITEC (Release 1.0) has had a major impact on the project. The design has been changed, a different host computer has been chosen, and schedules have been altered to make the portions needed for MER available earlier.

Work on the prototype MITEC systems in operation at Lincoln Laboratory and at Rome Laboratory has continued with the effort focused on providing input to the design of the production MITEC systems. A Bit Error Rate (BER) testing capability was added, waveform analysis and presentation were enhanced, and alarm polling and logging capabilities were extended. Software infrastructure improvements were carried out to support these new features, and the browse subsystem was enhanced. Changes were made in the testbed equipment at Lincoln Laboratory, and site visits were made and documentation generated in support of the testbeds at Rome Laboratory. Experiments explored alarm correlations, the effect of T1 jitter on equipment in the testbed, and the master/slave capabilities of the VF test sets.

Work was also carried out in the Expert Systems for Distributed Control (EDC) research area. The goal of this research is to determine how network control should be distributed throughout the hierarchy of a telecommunications system to maximize adaptation when coping with loss of resources or changes in requirements. The NETSIM simulation environment built to achieve this goal can now integrate simulations of the Transmission, Digital Patch and Access System (DPAS), and Defense Switched Network (DSN) levels of the communication system. A typical demonstration shows the effect of damage at the Transmission level on performance as seen by DSN users as well as the ability of the Autonomous Distributed Routing System (ADRS) algorithm to compensate for trunk outages by changing circuit routing at the DPAS level. NETSIM is also now integrated with the TRAMCON Alarm Interpreter (TAI) developed under DCEC sponsorship.

The MITEC work described in this report will culminate in completion of Release 1.0 during FY93. Lincoln strongly recommends that the preliminary design and engineering of Release 2.0 begin immediately, so that it will be possible to proceed with implementation of Release 2.0 immediately after Release 1.0 is delivered.

Table of Contents

Table of Figures

1. Introduction and Summary

This report describes work in three areas: the prototype MITEC systems in operation at Lincoln Laboratory and at Rome Laboratory; the production MITEC systems being developed for field deployment; and research in Expert Systems for Distributed Control (EDC). Work on the prototype MITECs is described in Section 2. It includes software development, testbed support, and experimental work aimed at providing input to the design of the production MITEC systems. A Bit Error Rate (BER) testing capability was added, waveform analysis and presentation were enhanced, and alarm polling and logging capabilities were extended. Software infrastructure improvements were carried out to support these new features, and the browse subsystem was enhanced. Changes were made in the testbed equipment at Lincoln Laboratory, and site visits were made and documentation generated in support of the testbeds at Rome Laboratory. Experiments explored alarm correlations, the effect of T1 jitter on equipment in the testbed, and the master/slave capabilities of the VF test sets.

Section 3 describes progress in the development of the production MITEC systems. It includes discussion of the decision to produce an Early Release version of MITEC in addition to the planned Release 1.0 version and reports on the design, progress, and schedules for both systems. The development plan involving the cooperative effort of three Air Force organizations, Rome Laboratory (RL/C3DA), the Air Force Communications Command Communications Systems Center (AFCC/CSC) and Technical Integration Center (AFCC/TIC), with Lincoln Laboratory and its subcontractor Structured Systems and Software, Inc. (3S) is also described.

Section 4 describes activities in the Expert Systems for Distributed Control (EDC) research area. The goal of this research is to determine how network control should be distributed throughout the hierarchy of a telecommunications system to maximize adaptation when coping with loss of resources or changes in requirements. The NETSIM simulation environment built to achieve this goal can now integrate simulations of the Transmission, DPAS, and DSN levels of the communication system. A typical demonstration shows the effect of damage at the Transmission level on performance as seen by DSN users as well as the ability of the ADRS algorithm to compensate for trunk outages by changing circuit routing at the DPAS level. Integration of NETSIM with the TRAMCON Alarm Interpreter (TAI) developed under DCEC sponsorship is also discussed.

Two appendices contain the Baseline Specifications for MITEC (Release 1.0) and MITEC (Early Release).

The MITEC work described in this report will culminate in completion of Release 1.0 during FY93. Lincoln strongly recommends that the preliminary design and engineering of Release 2.0 begin immediately, so that it will be possible to proceed with implementation of Release 2.0 immediately after Release 1.0 is delivered. There are three reasons for this recommendation:

i. Extensive as it is, the Release 1.0 Baseline Specification was deliberately limited to keep implementation cost and time within reason. As such it necessarily leaves out features that Technical Control personnel will doubtless want, such as categories of circuits and fault isolation procedures not yet in the Release 1.0 knowledge base. Technical Controllers will put up with these deficiencies for a while if they know that work is in progress to relieve them; otherwise, instead of realizing its potential to revolutionize Technical Control, MITEC will sink in their estimation to just another incomplete tool that helps with a portion of the work in a TCF.

ii. Technical Control is gradually changing. MITEC focuses primarily on low-speed circuits and tail circuits, while many modern TCFs deal almost exclusively with T1 and above, and with programmable muxes such as the DPAS and the IDNX switches of the developmental AFNET, and have few if any low-speed circuits. Hence Release 1.0 will be simply unuseable in a large and growing fraction of TCFs.

iii. It is very important to avoid letting MITEC go the way of the ATEC (Automated TEch Control) project of some years ago. ATEC failed because it had a number of serious deficiencies in its initial release, and there was never a follow-up to correct them.

The philosophy of reason ii was clearly supported by a trip report filed by AFCC/CSC personnel describing a mid-September 1991 visit to four European TCFs to obtain user community input on the needed form and features of MITEC. The report noted that many European TCFs have little need for the AN/FCC-100 LSTDMs and low-rate circuits featured prominently in MITEC (Release 1.0); a number of senior Technical Controllers told the AFCC visitors that they will depend heavily upon DPASs. To further motivate this work, consider that the AFNET which is currently in the acquisition process has an open requirement for system control functionality, which translates directly into a need for AFNET network control systems applicable to IDNX networks and amenable to operation by Air Force enlisted personnel.

From the above considerations it appears that there is a clear need for a MITEC (Release 2.0) which:

i. Incorporates the low-speed circuit control capabilities of Release 1.0;

ii. Provides an intelligent user interface and automated capabilities for managing IDNX and similar reconfigurable transmission systems;

iii. Fully integrates the Technical Control of both tail circuits and long-haul transmission for TCFs that have both functions; and

iv. Provides for activation of only the low-speed functions, or only the long-haul functions, at TCFs which have only one role.

2. Prototype MITEC

The term "prototype MITEC" in this report refers to the MITEC systems running in the testbeds at Lincoln Laboratory and at Rome Laboratory. These systems run in Symbolics computers and are coded in LISP. They have been under development for several years and were successfully demonstrated in the Washington area in FY90. In the current fiscal year effort has focused on the development of field deployable versions of MITEC called "production" MITECs (see Section 3.), but work has continued on the prototype systems. Most of the work in this year has been directed toward areas where experience with the prototype systems can contribute to the design and/or implementation of the production systems.

Work on the prototype MITECs has involved software development, testbed support, and the carrying out of experiments. These activities are reported in the following subsections. In addition, two "white papers" were generated. One deals with the issue of prompts from devices in response to commands from MITEC. The other is an annotated section of LISP code which embodies the Fireberd BER test algorithm in the MITEC prototype. Together, these documents are part of a series of "lessons learned" from the development of the MITEC prototype at Lincoln Laboratory intended to be of assistance to the developers of the production MITECs.

2.1 Software Development

During the year the prototype MITEC software was extended to incorporate bit error rate testing. Waveform analysis and presentation were expanded, and the software for alarm polling and logging was reworked and extended. Some needed improvements were made in the software infrastructure, and the browse subsystem was enhanced. These activities are reported in more detail in the following subsections.

2.1.1 Bit Error Rate Testing

The Fireberd 6000 Communications Analyzer has been selected as the bit error rate (BER) test instrument for use in the production MITECs. To aid in understanding its capabilities and to determine if there were any potential surprises that might turn up during implementation of the production versions, we decided to incorporate it into prototype MITEC browsing and diagnosis.

Our initial focus of attention dealt with the characteristics of communication between the computer and the Fireberd using ASCII over an RS-232 line. This communication is used for sending remote commands to the Fireberd and receiving replies to the commands. A secondary initial focus was to determine the commands that are needed to initiate tests and to obtain results.

In order to satisfy MITEC's command-response paradigm we needed to determine the Fireberd 6000's prompt that is issued when it is ready for another command. (Note that a command line to the device can consist of a single command or several commands separated by semicolons.) After much investigation we realized that the device issues two kinds of prompts: (a) a prompt consisting of ">" after it has finished analyzing each command on a command line, and (b) a prompt consisting of a NUL character (all zeros) followed by ">" when it is completely done with its processing of all commands on the command line. It is the condition of the device being done with its processing that is of importance to MITEC, but the latter prompt is generated only if the command(s) produce typed output. Therefore, all command lines to the device need to conclude with a (possibly superfluous) command that produces some output. This is not a serious problem; it is easily handled by appending an appropriate command whenever necessary. More serious is the lack of such a prompt whenever the entire output for a command line consists of error messages.

Another potential problem is the handling of NUL as a character in a prompt. This non-printing character may be stripped or otherwise treated specially by operating systems or languages such as 'C' which uses it as a string delimiter. We had to make changes in our character input software to deal with it successfully. Its use by the Fireberd manufacturer was an unfortunate choice from the point of view of programmers implementing software intended to operate the instrument automatically.

A BER test involves the transmission of a known test pattern at one end of the circuit being tested and the receipt of that pattern, possibly corrupted by errors, at the other. In the case of a loop test, the same instrument can be both transmitter and receiver. At the receiver, the first step in a test is that of acquiring synchronization with the transmitted pattern. Our implementation handles synchronization in two situations. When the test begins, we enter a tight loop waiting for synchronization between the two ends. If such synchronization is not acquired by a certain specifiable time, the test is deemed to be a failure and is so indicated. Later on, during the test, polling is done every few seconds in order to check on progress. If a synchronization loss is reported during any of the polls, the test is immediately halted, and a report is generated consisting of the results (bit and block errors, etc.) up until the time at which sync was lost. The duration of the test prior to sync loss is also reported. The MITEC operator has menu items which provide for the choice of the length of time to wait for synchronization to be acquired and the length of time to run the test after acquisition.

We incorporated the Fireberd 6000 capabilities into fault isolation in two areas of specific interest to the production MITECs. First, the prototype MITEC now troubleshoots poor-signal complaints in addition to no-signal complaints. By degrading a circuit line's quality via a phone line simulator, we create a user complaint of poor-signal quality which MITEC troubleshoots using the Fireberd 6000. Second, we demonstrated that, by using the Fireberd 6000 in conjunction with specific modem installation standards, we can command the Fireberd 6000 to establish a digital loopback on the Codex 2510 modems used in our tail circuits. This capability enhances the troubleshooting that can be done on tail circuits without remote line units. (The Codex 2510 modem can easily be put into digital loopback mode from its front panel or through the Codex modem management system, but neither of these two methods is viewed as appropriate for MITEC.) MITEC (Release 1.0) will use remote line units in troubleshooting tail circuits. Adopting modem installation standards so that the Fireberd 6000 can loop back the Codex modems is an option for future releases.

2.1.2 Waveform Analysis and Presentation

Waveform analysis has been expanded in two major areas: distinguishing between the signal inside the matrix switch and the one presented at the monitor port; and handling a wider variety of signal types and characteristics. A situation in which these aspects of waveform analysis become important is illustrated in Figure 1 which shows the possibilities for accessing the FCC-100 aggregate signals between that multiplexer (mux) and the FCC-98 mux that carries the aggregate signal on one of its channels. In the figure, the boxes labeled 'DTE' and 'DCE' are ports on the matrix switch. The labeled arrows indicate the types of signals passing in the two directions. The dashed lines from the scope indicate the two possibilities for observing waveforms. At TP1, the scope sees the output of the matrix switch monitor port. At TP2, it sees the true signals passing between the matrix switch port and the FCC-100. These signals are balanced RS-449 and MIL-188 in the receive (RX) and transmit (TX) directions, respectively. TP1 and TP2 correspond to access points on the Hekimian 3200 access switch which provides metallic connections to the signal wires.

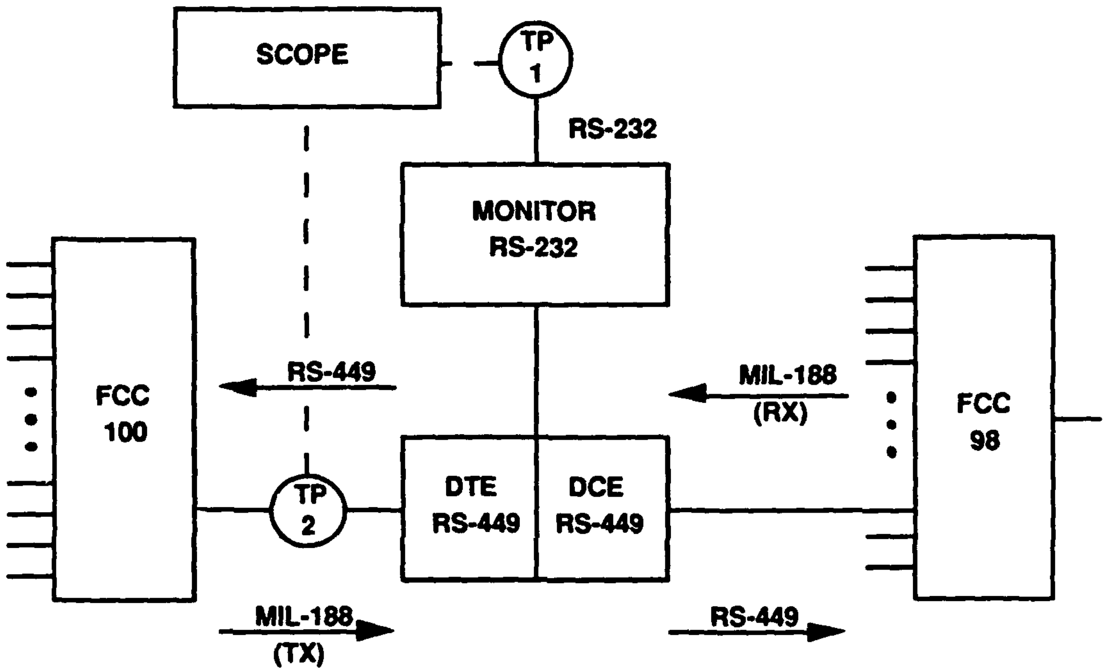


Figure 1. Scope Monitor Access Example

The Telenex Mini-Matrix switch in the prototype testbed converts incoming digital signals into internal proprietary forms and then reconverts to the appropriate signals on output. These output signals are either RS-232 or RS-449. The internal conversion involves a certain amount of "cleaning up" of the signal by considering the signal's value at each point as either mark or space. This internal signal in the switch is not directly observable by the digital oscilloscope. Instead, we can observe signals going through the switch only via an RS-232 monitor port which taps into digital circuits passing through the switch. These limitations on observation have many consequences. For example, when the object signal is RS-449, the monitored output shows the mark-space voltage and the rise-fall times of an RS-232 signal. Even if we had an RS-449 monitor, we would not be observing the true output of the matrix switch. Any monitor port presents an independent reconstruction of the signal waveforms from the internal representation.

In the past we analyzed and presented the RS-232 monitor output to the MITEC operator without further comment about internal switch signals. Since this can be misleading to the observer, we now normally present an "inside-switch" display which is a reconstruction of the mark-space form of the signal that we presume is present inside the switch. This display is generated from the monitor output by smoothing out the values at a mark or space and converting the values observed during rises or falls into the marks and spaces toward which they are heading. As a result, the display is rectangular with instantaneous rises and falls. Grid marks and value labels along the vertical axis are replaced by 'mark' and 'space' labels. While we normally present this idealized inside-switch waveform to remind the operator that he is not seeing a "real" waveform for which mark and space voltages and rise-fall times would be meaningful, we also have available for display the actual monitor output and a superimposition of it with the inside-switch waveform which can be called up from menu selections.

The distinction between the inside-switch and the monitor output has led to an ordered analysis of a waveform. First, the quality of the monitor signal is checked for reasonable mean values, rise-fall times, and other basic characteristics. If this check fails, then the monitor output is deemed to be defective, no further analysis is carried out, and the monitor output is presented with this judgement. If the preliminary analysis of the monitor output passes, then the inside-switch form of the output is reconstructed and analyzed; both the reconstructed waveform and its analysis are then presented.

The above distinctions between monitor output and inside-switch output do not apply in cases where the signal can be observed directly by the scope without the use of the matrix switch, e.g. at TP2. In such cases, there is no need to reconstruct a signal, and therefore only one presentation is available, and only one analysis is provided.

The MITEC waveform software was further extended to handle direction-dependent parameterization and balanced vs. unbalanced properties of signals. This case occurs in the situation shown in Figure 1. There the aggregate side of an FCC-100 is connected through the matrix switch to a port of an FCC-98. The FCC-100 outputs balanced MIL-188 towards the switch, while the switch outputs balanced RS-449 towards the FCC-100. If MITEC is directly observing the signal between the switch and FCC-100 with one scope channel on each of two signals of interest, e.g, data and clock, then it is faced with different characteristics in the two directions as well as making correct judgements of signal quality from the observation of only one half of a balanced signal in each case. If our scope had four-channels, the two signals could be examined in balanced form, but it has only two, and the software must cope with the limitation. For example, mark-space for half of a balanced MIL-188 ranges between positive and negative values while for half of the balanced RS-449 output of the matrix switch, the values are always positive. To handle such diversity on the two scope channels, the analysis is parameterized independently for each. The required information is obtained by the waveform analysis program from the database, aided in part by the new monitor objects that have been added to the software infrastructure (see Section 2.1.4 below).

2.1.3 Alarm Polling and Logging

The software which polls the Datalok for alarms has been reworked and expanded. We now obtain the alarms via bit maps. The alarm information is stored and characterized according to device and when the alarm last changed. As many as 240 different alarms could now be handled, but our hardware configuration is currently limited to 36 alarms. When browsing, one can request that a poll for alarms be performed and that the results be presented in any of several different formats. These new capabilities have been used in the knowledge engineering of alarm behavior resulting from various faults, as described in Section 2.3.1 below.

We corrected some problems in the automatic fixed-interval polling of the Datalok for alarms. When a poll detects an alarm that is suitable for MITEC's fault isolation methodologies, a new diagnosis is optionally started. The strategy for evaluating competing potential diagnoses (as obtained via Datalok polling, user complaint, or request from another TCF) in order to determine their execution priority is currently being explored.

In anticipation of experiments to be performed with microwave links at Rome Laboratory, we designed the software to have a mode in which the alarms detected by the Datalok 10A are logged into a disk file. This time-stamped information is logged in binary format permitting a short, fixed-length record corresponding to each poll for alarms. A simple decoding program reads and decodes the log file. This entire facility has potential for life studies of failures, searches for failure combinations, and quality control studies. Its direction awaits users.

In polling the Datalok for alarms on a 30-second cycle, one typically obtains sequences, sometimes fairly long, of identical reports. In order to decrease disk space usage, an alarm report is written to the disk only when it differs from the previous report. If, as a result, nothing has been written in an hour, then a report is written even though it does not differ from the previous one. Such an hourly checkpoint provides confidence that the alarm polling function is working. In case of a system crash, the hourly checkpoints provide alarm information to within an hour of the crash. These features can result in saving a substantial amount of disk space, depending upon the frequency of changes in alarm state. In no event, however, will more disk space be used.

The software for decoding the disk files produces easy-to-read output, with each alarm printout containing a date-time, the alarm name, the device involved, and the Datalok's alarm number for the alarm. The format is organized to make it convenient to implement further processing of the output, if such should be desired in the future.

2.1.4 Software Infrastructure Improvements

A queueing mechanism has been added to the MITEC Dispatcher, the MITEC process which communicates with devices via the TODOWN program on the Sun . The queueing mechanism enables MITEC processes to concurrently use different logical devices which map to the same physical device. An example of such a device is the Datalok that both detects alarms and drives the scanner that selects pins for waveform measurements.

If two processes try to communicate concurrently with the same physical device, the queueing mechanism makes sure that only one command to the device is outstanding at a time. Other commands to the same physical device are queued and sent on a FIFO basis. This queueing mechanism does not protect against one process overriding the device state that another process expects. Such protection is achieved by scheduling the entire physical device. For devices that can be meaningfully split into logical components, there is no overlap between the state information applicable to the logical components, and so the scheduling of the logical device will suffice.

Command queueing allows the Datalok to be defined as two logical devices: an alarm detector and a scanner for selecting waveform pins. Similarly, it permits the Telenex Mini-Matrix switch to be defined into as many logical devices as there are monitor ports. When a MITEC process requires a circuit measurement, the monitor port will be tied up longer than the actual time that the matrix switch control port is required. By partitioning the matrix switch into multiple logical devices, MITEC processes using the matrix switch are interleaved more efficiently.

We have implemented the infrastructure required for interrupting a fault isolation process. The motivation for this feature is the goal of the prototype MITEC to correlate and prioritize complaints received from multiple sources: alarms, users and remote TCF requests. When two or more complaints which relate to the same fault are received at different times, then MITEC may need to suspend a fault isolation process currently underway and focus on a higher-level problem.

Upon determination that a fault isolation process (P) should be interrupted, a filtering process initiates the halting of P. If a flag indicates to the filtering process that P cannot be interrupted at the moment, then P is told via another flag to halt itself at the first available opportunity. Even at that point, P may be in the middle of a sequence of operations such that halting would cause one or more devices to be left in an unclean state. We have made some progress in tracking down and executing the required cleanups for processes that are halted before normal completion, but more work is needed in this area.

In order to support the flexible use of MITECs measurement equipment, we created the concept of a Monitor Object (MO) which is a software infrastructure element with two major characteristics: a Function and an Appearance. The MO's Function describes its appropriate use. An MO can be used to retrieve a waveform, perform a BER test, or assess VF signal quality. The MO's Appearance describes where the signal to be monitored can be found. For example, in the case illustrated in Figure 1 of Section 2.1.2, the MO will provide the information needed by the waveform analysis program to set the Telenex monitor and Hekimian 3200 access switch properly, and to assist it in finding the correct signal parameters needed to assess the signal quality.

Monitor Objects have provided a level of abstraction needed in order to test circuits appearing at either Hekimian 3200 access points or Telenex matrix switch ports with the same test equipment. One of our testbeds is wired so that the Telenex monitor ports connect to Hekimian 3200 access points, and the test equipment connects to Hekimian 3200 test appearance lines. The other testbed has the Telenex monitor ports directly connected to the test equipment. The MO abstraction makes the rest of the software transparent with respect to this level of complexity.

After creating the concept of a Monitor Object, we implemented the necessary primitives to access a circuit correctly. We introduced inheritance into access points; some signal characteristics seen at access points are based on the access point itself, while other characteristics are based on the signal passing through the access point.

The test and access configuration in the prototype MITEC minimizes the number of test instruments absolutely needed by a MITEC with multiple access switches, since the same instrument can be used on either switch. The anticipated software complexity of defining and managing this configuration led to the conclusion that MITEC (Release 1.0) should be configured to have independent sets of test instruments on each access switch. Our experience with Monitor Objects suggests that this conclusion should be reconsidered for future MITEC releases.

The creation of Monitor Objects effectively partitioned the matrix switch into schedulable units which can be used concurrently. This feature, in conjunction with command queueing by the Dispatcher, enables processes to use matrix switch monitors without blocking other processes from using the matrix switch during the same time period.

2.1.5 Browse Subsystem Enhancements

One may now request an iterative browse of all the items available at a certain stage of the Browse subsystem. For example, the top level Browse menu presents a list of all the types of items that are available in the MITEC database. Selecting "all" from this menu results in a succession of menus, one for each type of item. These second level menus permit one to choose a specific item to browse. Selecting "all" at this point produces a succession of browses of all the items of that type. Further, when one is browsing a circuit, one can choose "browse all" and successively browse all the components that are currently shown in the "level b" display of the circuit.

When browsing an FCC-100, one has available several new capabilities. One is now able to Examine Active (or Offline) Aggregate and obtain Status Errors. One may also make changes by Configure Port (or Aggregate) and by Activate Local. All of these operations are available via menus, which is more convenient than typing commands directly to the FCC-100.

The software that provides browsing of the Telenex Mini-Matrix Switch has been expanded significantly. Several new commands provide new forms of output and enable one to reconfigure the Switch.

An operator may now request output that compares the Switch configuration to the MITEC database information. The operator is thereby able to determine at a glance how the Switch mappings differ from the prescribed ones. Another output format shows the current mappings in the Switch and the devices connected via these mappings. For example, this output would show for a DCE-x to DTE-y connection the devices (and the appropriate ports) that are connected to DCE-x and DTE-y. If a null modem is involved, that is also shown. In this way, the output decodes the Switch mappings into meaningful connections between devices.

Using the mapping information just described, the operator may elect to reconfigure the Switch. To assist in this effort, a new set of commands provides for making and breaking Switch connections. These commands prompt for the parameters involved in altering connections and provide a complete list of the repercussions that would result from the changes. After viewing the list, the operator can continue with the change or cancel the request. If the operator elects to continue, the software issues the appropriate low-level commands to the Switch. With this new facility, the operator no longer needs to learn and type low-level commands directly to the Switch.

2.2 Testbed Status

2.2.1 Lincoln Laboratory Testbeds

Several changes were made throughout FY91 in Lincoln's testbed hardware and configuration. The changes were as follows:

Two borrowed FCC-100s were returned to DISA. The remaining two FCC-100s were upgraded from Version 1 to Version 4.

The TDM-153 Channel Banks were removed from the primary testbed configuration and replaced with FCC-98s. The Channel Banks were then connected through the DACS for use in experiments.

The FCC-98s were reconfigured for loop-back timing and the DACS was connected between them in master timing mode, as preparation for experiments in error analysis.

One Hekimian 3200 Test Access Switch was removed and sent to CSC/SDCE to support the development of the production MITEC software.

We experienced several hardware failures in the testbeds:

One FCC-98 failed power supply was replaced.

One Datalok 10A failed power supply was repaired.

One Hekimian 3705 VF test set power supply was replaced.

One Datalok 10A board was repaired to correct an alarm sensing problem.

Analysis of apparent incompatibilities between the Hayes 9600 bps modems and the VF line through the Telenex matrix switch continues.

2.2.2 Rome Laboratory Testbeds

At the beginning of FY91, the installation and expansion of the MITEC testbeds at Rome Laboratory neared completion. Additional activities included: establishing an easier process of bringing up MITEC on the Symbolics machines, generating new illustrations of the testbed configuration, writing textual summaries of demo scenarios, and working on a manual for maintaining the MITEC database.

New on-base and long-haul databases were subsequently developed and tested for each testbed. We implemented a third database configuration for the testbeds which incorporated the additional multiplexers. This latter database reconfigured the circuits to consist of three levels of multiplexing.

On 9-10 January a trip to Rome Laboratory was made to configure and test their newly-installed DACS. Testing was successful, but a critical oscillator, temporarily borrowed from Lincoln for testing, had to be ordered from the manufacturer in order to make the DACS at Rome fully operational.

At the end of January, Mr. Scott Rabe of Rome Laboratory accompanied Capt. Morse and Capt. Hatten to Lincoln Laboratory. During that visit, Scott familiarized himself with the rudimentary workings of the database and discussed future plans for the testbeds with various Lincoln staff members. It was planned that Scott will be able to add various circuits and pieces of equipment to the Rome Laboratory testbeds and make the associated necessary changes to the MITEC database. As an ongoing effort, we have provided assistance to Rome Laboratory personnel in supporting their demo-presentation efforts and in their database modifications.

Work began on a manual for maintaining the MITEC database. The manual begins with an overview of the LISP concepts of flavors, mixins, and instance-variables. Two brief sections follow, one on locating the proper directory, subdirectory, and file, and another with general information about the format of the database entries. The main part of the manual will be a series of chapters to be used when adding or modifying equipment in the MITEC testbeds. These chapters will detail the correct format and acceptable instance-variables and values for entries in the database. Three sections have been completed so far, one each for electronic-patchboxes, circuits, and trunks. A copy of the manual as it currently stands was given to Mr. Scott Rabe in April.

During the spring, the Rome Laboratory testbeds, "Pentagon" and "Griffiss", were temporarily rendered inoperative when DISA reclaimed its hardware from the Pentagon testbed. However, by utilizing locally available spare equipment, Pentagon was substantially rewired to its previous configuration, lacking only a few essential pieces of equipment. Arrangements were made to purchase or borrow these in order to make both Rome Laboratory testbeds fully operational again.

In June, the system software was restored on one of the Symbolics machines following a disk repair. The latest versions of the MITEC software were installed on both Symbolics, thus, removing the dependency on the unreliable Ethernet connection between the two machines. A malfunctioning terminal was replaced on one of the testbeds. A limited demonstration capability was reestablished.

On 10-11 September, a trip to Rome Laboratory was made to investigate two testbed problems and provide general support. The first testbed problem showed up as waveforms inconsistent with a circuit's behavior. The problem was related to the installation of a new Datalok 10A which was improperly configured. The second problem was observed when using any but the first monitor port on a Telenex matrix switch. This was again related to the installation of new equipment. It turns out that a new Telenex board which addresses the monitor ports differently from any of the other matrix switches had replaced one of the boards returned to DISA. Additionally, we worked with Mr. Scott Rabe on various demo scenarios.

Currently, the Rome Laboratory testbeds are operational and a demonstration capability is restored. Two BERTs and a Telenex switch shelf are still needed to achieve full capacity.

2.3 Experiments

2.3.1 Alarm Correlation Experiments

We completed a series of experiments aimed at providing alarm correlation information that will be useful for the production MITEC development effort. The experiments were performed using the facilities of the Lincoln MITEC testbed. They involved the DACS plus the FCC-98 and FCC-100 multiplexers. In early experiments, two levels of FCC-100 multiplexing were used, but with the return of the borrowed FCC-100s to DISA, later experiments were limited to one level.

The occurrence of alarms is observed by looking at front panel lights, by requesting alarm status reports from the FCC-100s and the DACS, and by polling alarms through the recently enhanced Datalok software in MITEC. The effects of the signal degradations that may or may not result in alarms are also observed by looking at the bit errors detected by the BERTs that serve as users for the testbed circuits.

The focus of early work in this area was on introducing problems at the T1 level and observing their effects at lower levels. The configuration used involves splitting the single DACS into two logical units with a T1 span between the two. Thus there are three T1 spans between the East and West TCFs: two FCC-98-to-DACS spans and one DACS-to-DACS span.

We have two instruments capable of introducing problems into a T1 span. One is an attenuator. The other is the Fireberd 6000 which, with the options originally ordered, is capable of introducing jitter, bipolar violations, CRC errors, and framing errors. Unfortunately, without an additional option, it can introduce these problems only in conjunction with its own test signals, making it useless for correlation experiments because the lower level multiplexers must be disconnected. The Fireberd option to allow the problems to be introduced while passing other data is relatively expensive, but it can be rented, and we chose to do so for a short time in order to carry out a set of experiments with these kinds of problems. The experiments described in Section 2.3.2 below made use of this rented option.

In our first alarm correlation experiments we used the attenuator to reduce the amplitude of the received bipolar signal. We found that in the DACS-to-DACS span there was essentially no effect until the attenuation was so great that complete loss of the span was registered. (The attenuator operates in 1 db steps, and the DACS goes from near-perfect operation to complete loss of communication in one step.) With the attenuator in the path to one of the FCC-98s, however, we could generate a range of effects between normal operation and complete failure. With modest levels of attenuation, we observe bit errors at the user BERTs but no detected alarms anywhere in the transmission equipment. With somewhat more attenuation, the FCC-100s exhibit intermittent Loss-Of-Frame (LOF) alarms at one or both multiplexing levels. (If the higher level mux has LOF, the lower level also has the same condition, but the lower level mux can show LOF without any corresponding alarm at the higher level.) Of course, even a momentary LOF in the mux causes a large number of bit errors and very likely a loss of sync to be seen at the BERTs. In this range of attenuation, there are no alarms registered at the FCC-98s.

At higher attenuations, the T1 level mux (either DACS or FCC-98) will show alarms such as the Carrier Group Alarm (CGA) that indicate failure of the span. (The FCC-98 has three such alarms, but we have not been able to obtain any combinations except all or none.) In the CGA situation, both lower-level FCC-100s show LOF, and the BERTs show loss of signal. This situation is the same independent of which T1 span shows the CGA and in which direction the attenuation is introduced. If the CGA occurs in the DACS-to-DACS span, no alarms appear at the FCC-98s.

In the course of the early experiments we had occasion to observe the differences in the ways that the two versions of FCC-100s reported the same alarms and other conditions. We circulated our observations of those differences to the other organizations involved in the production MITEC development effort.

It is clear from our experiments that the rate at which FCC-100 alarms can show state changes is very much higher than the alarm polling rates that have been considered for the production MITECs. Further work is needed to assess the potential for alarm inferencing in a situation where input information is distorted by slow sampling of changing signals.

A conclusion drawn from the experiments is that we should make similar tests using borrowed or rented instrumentation as part of the production MITEC test procedures.

A side result of our alarm experiments was the discovery of a situation in which a LOF condition in the Version 1 FCC-100 correctly registered on the alarm relay sensed by the Datalok 10A and on the front panel alarm light, but not in the status reported on the front panel LEDs or via the RS-232 terminal, which indicated normal operation. Repeating the experiment after installing our recently received Version 4 upgrade kits showed that while the front panel LEDs still erroneously displayed operation as normal, a status query from the RS-232 terminal now correctly showed a LOF status. This is good news for the production MITECs which will be making use of such status information.

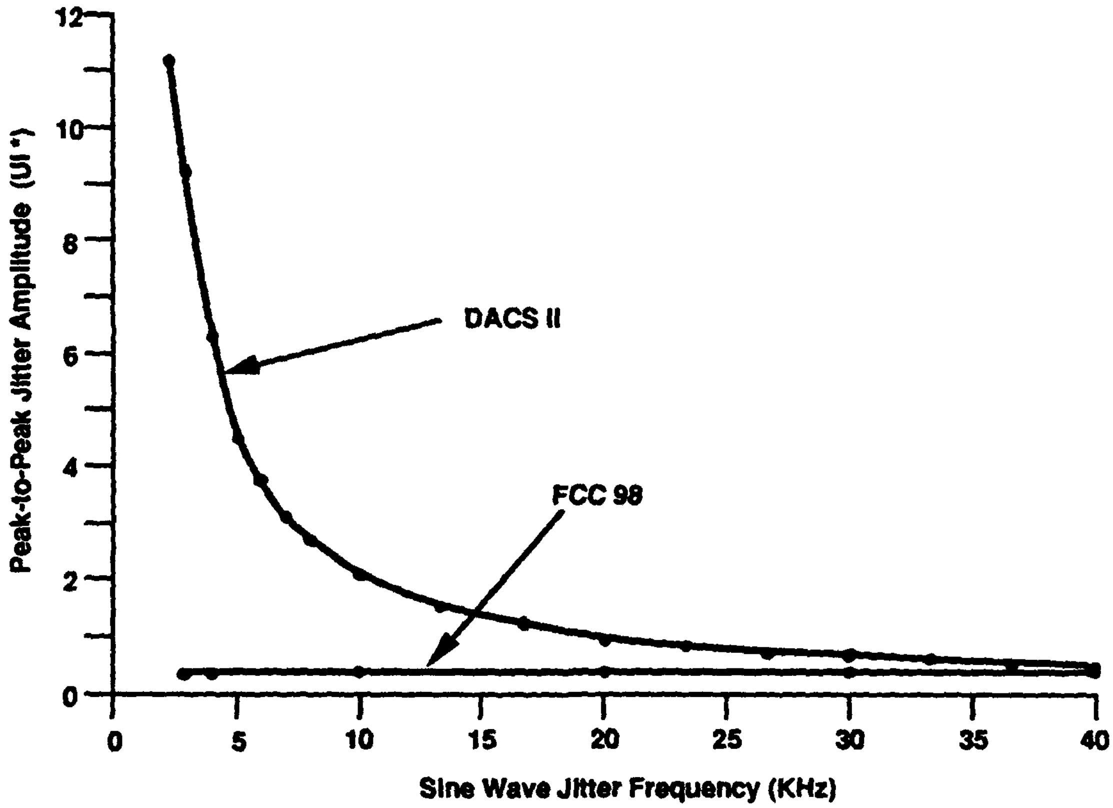
2.3.2 T1 Jitter Sensitivity Experiments

Preliminary experiments using the Fireberd 6000 to introduce jitter in the T1 signal show that the intermittent LOF effects described in the previous Subsection can also result from jitter. As for the case of attenuation, the FCC-98s show more sensitivity to jitter than does the DACS.

Our work on the effects of jitter introduced into T1 spans made use of the Fireberd 6000 with an optional DS1/T1 Data Interface (Model 40540) rented for six weeks to support the experiments. This interface allows jitter to be introduced in a signal passing through the Fireberd 6000. Without the special interface, the instrument can introduce jitter only on its own test signals. The instrument allows the jitter amplitude, frequency, and modulation type to be varied. In our experiments, we used a sine wave modulation and varied amplitude and frequency. Jitter was introduced both in a DACS-to-DACS span and a DACS-to-FCC-98 span.

There were no real surprises in the results. The DACS was found to be much more tolerant of jitter than the FCC-98 as illustrated in Figure 2 which shows the threshold conditions at which jitter caused frame loss (failure) of the T1 span. The curves show that the DACS can follow timing changes in the signal of many times the nominal interval between pulses as long as the changes occur relatively slowly (rates of a few KHz, only). By contrast, timing shifts of only a fraction of a bit time at any rate measured caused problems for the FCC-98.

In general, as jitter amplitude and/or frequency were increased from zero, bit errors began to be observed at the user BERTs in the testbed. The error rate increased with the jitter. At higher levels of jitter, loss-of-frame alarms were observed, first in the FCC-100 multiplexers and eventually in the FCC-98s and the DACS. When the jitter took place in a span to the DACS, the expected correlation was observed between the rates of bit errors and low-level mux alarms and the rates of bipolar violation, change-of-frame-alignment, out-of-frame, and frame slip events reported by the DACS.



* One UI (Unit Interval) equals 648 ns., the time associated with one T1 bit period.

Figure 2. Thresholds for loss of T1 frame due to jitter.

2.3.3 Voice Frequency Test Set Experiments

A series of experiments was conducted with the Hekimian 3703 and 3705 voice frequency test sets to explore their master/slave capabilities. This capability allows a unit at one end of a circuit to be tested to operate a remote unit at the other end as a slave. The slave unit both provides test signals and makes measurements, allowing the circuit to be tested in both directions with one procedure. Since our units are not equipped with identical options, we were unable to carry out some procedures of interest for the production MITECs, but we were able to show that the master/slave operation proceeded quite smoothly. The required commands to capture and release the units from slave mode were noted. In addition, attenuation and noise from the telephone line simulator were introduced into the line to determine at what point the communication between the 3703 and the 3705 began to deteriorate. The software in the production MITECs will have to be written to cope with occasional instrument-to-instrument communication problems caused by noise on the line under test. Our results and observations were sent to the other organizations working on the production MITECs which will use similar instruments.

3. Production MITECs

In FY90 a Memorandum of Understanding was signed among Rome Laboratory (then RADC), AFCC, and DISA (then DCA) providing for joint support of an effort to develop a field-deployable version of the prototype MITEC system that had been successfully demonstrated in the Washington area earlier that year. The name MITEC (Release 1.0) was chosen for the deployable system, and a plan was worked out for the development of the software in Ada according to DoD Standard 2167A as a joint effort between Lincoln Laboratory and a group at the Air Force Communications Systems Center (CSC/SDCE, then CCSC/COE). Structured Systems and Software, Inc., (3S) of Laguna Hills, California, a subcontractor to Lincoln Laboratory with experience in the development of Ada software, began work in late FY90 with funding from AFCC. A Baseline Specification was written for MITEC (Release 1.0), and an official start-of-effort took place on 1 October 1990, with Rome Laboratory (RL/C3DA) acting as Program Management Office. A target delivery date for MITEC (Release 1.0) was set for early FY93.

At the MITEC Program Management Review held at Lincoln Laboratory on 18-20 December 1990, Colonel Mittelmann, HQ AFCC/PG, made a strong recommendation that some portion of the MITEC capability be delivered substantially earlier than the 1Q FY93 target for MITEC (Release 1.0). During the wrapup of the meeting, Captain Flynn of AFCC asked that written proposals, including a description of the delivery and estimates of cost and schedule impact be sent to him and Colonel Mittelmann for review. By mid-January a decision was made to combine some MITEC functionality with the DCEC-sponsored Technical Control Automation Project (TCAP) system scheduled for delivery in February 1992. A proof-of-concept demonstration of the TCAP system had been conducted at Ft. Detrick in FY90, and a rehosting of the system on the Air Force standard Desktop III computer was under way at Sandia National Laboratories. TCAP has two main features: administrative database and report generation, and remote control of FCC-100 multiplexers. It was decided to integrate into TCAP the MITEC modules for controlling the VF and digital test equipment planned for MITEC (Release 1.0). The resulting merger would enhance TCAP and would, through the MITEC-style human machine interface (HMI), provide MITEC visibility in the desired time frame. The name MITEC (Early Release) was chosen for the MITEC portion of the combined system.

The decision to develop MITEC (Early Release), also referred to as MER, in parallel with MITEC (Release 1.0) has had a major impact on the project. The following sections describe the two systems, FY91 progress, and the status of each at the end of the year. We use the term "production MITECs" to refer to both systems. Hence the overall heading for this section of the report.

3.1 MITEC (Release 1.0)

3.1.1 System Overview

The function of MITEC (Release 1.0) is to assist Technical Controllers by detecting equipment alarms, troubleshooting faults and restoring communication on Defense Communications System circuits managed by Technical Control Facilities (TCFs).

The functional characteristics of MITEC (Release 1.0) are identified in the MITEC Baseline Specification document. The Baseline Specification, included as Appendix A, states that system development will conform to DoD Standard 2167A. According to that standard, the System Design Document (SDD) establishes the system-level design as it relates to the technical control mission and operational environment. The SDD is traceable to the Baseline Specification.

The SDD specifies that MITEC (Release 1.0) will consist of the following configuration items: a Hardware Configuration Item (HWCI) made up of thirteen Hardware Inventory Items, two developmental Computer Software Configuration Items (CSCIs) named Core and Equipment Interface (EI), and five off-the-shelf Non-Developmental Configuration Items (NDCIs). The SDD describes the architecture of the HWCI, the role of each NDCI required by MITEC, and the designs of both CSCIs. The SDD was officially issued on 14 January 1991. Revision A was issued 30 September 1991.

MITEC may conveniently be thought of as consisting of three groups of components: the MITEC host computer and the software residing thereon; the test and communications equipment controlled by the host computer; and the Technical Controller terminals. These components and the interfaces between these components are illustrated in Figure 3. The MITEC host computer controls the test and communications equipment via standard EIA-232 and IEEE-488.2 (GPIB) interfaces. Technical Controller terminals (up to 4) are connected to the MITEC host computer via an IEEE-802.3 Ethernet interface.

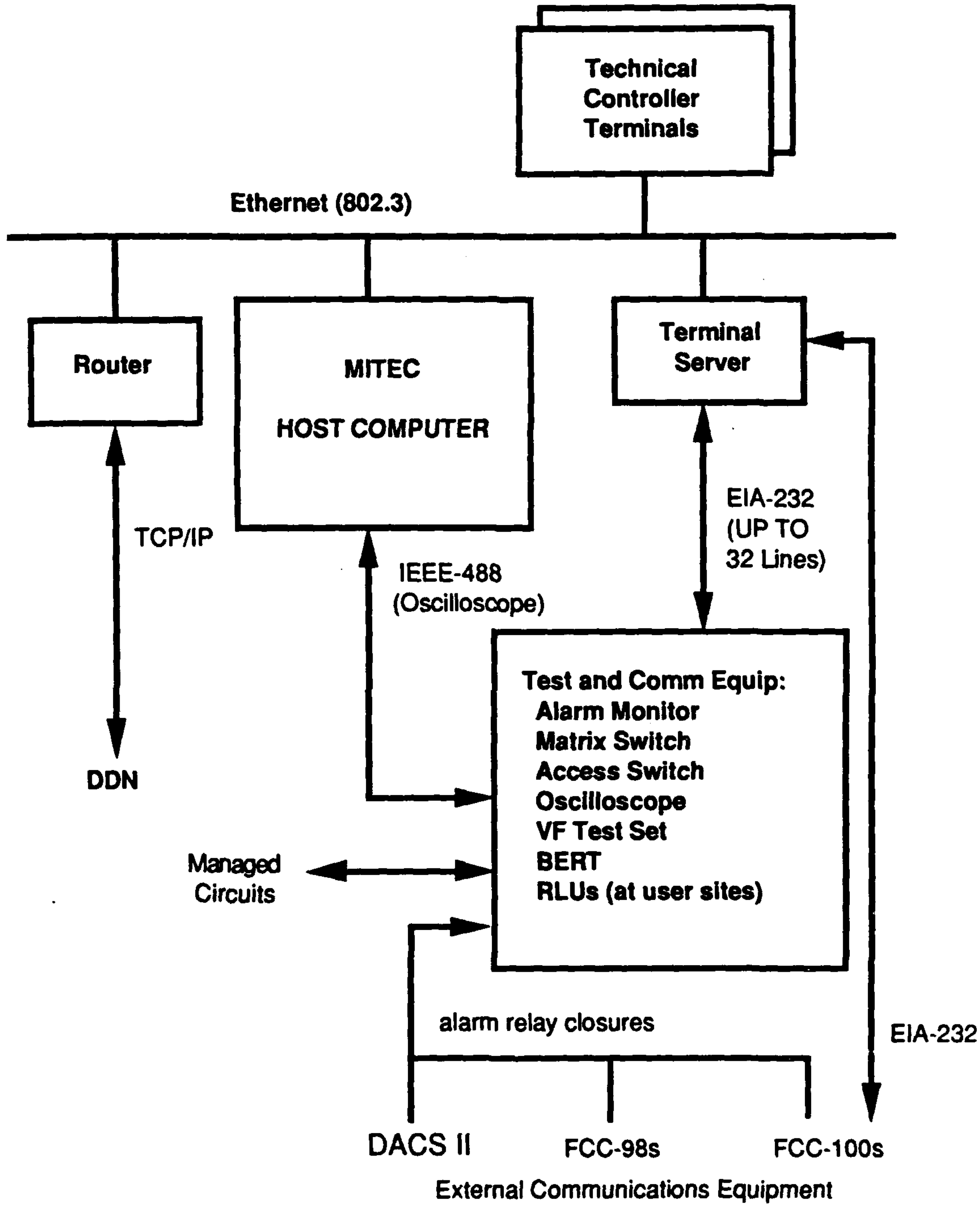


Figure 3. MITEC (Release 1.0) System Diagram

External interfaces of the MITEC system are also illustrated in Figure 3. The Technical Controllers interact with MITEC via the Technical Controller terminals. The MITEC Host Computer controls and monitors external communications equipment via standard EIA-232 interfaces. The Router provides an interface to DDN for communication between MITEC installations. The Test and Communications Equipment interfaces to the circuits managed by MITEC. This equipment is capable of testing circuits in order to diagnose faults and rerouting them to restore service. The Test and Communications Equipment includes alarm monitors that gather alarm data from external equipment allowing MITEC to detect and infer circuit outages.

3.1.2 Development Plan

The following organizations are involved in the development, support, and operation of MITEC:

a. Rome Laboratory (RL/C3DA), Griffiss AFB, NY. RL/C3DA is the MITEC Program Management Office (PMO) and is responsible for oversight of the MITEC program throughout development.

b. M. I. T. Lincoln Laboratory (LL), Lexington, MA. LL through its subcontractor Structured Systems and Software, Inc. (3S) is responsible for the development of the MITEC Core CSCI.

c. Communications Systems Center (CSC), Tinker AFB, OK. Two organizations within CSC, SDCE and SDQA, are involved in MITEC.

   CSC/SDCE is responsible for the development of the MITEC Equipment Interface (EI) CSCI and the integration of the MITEC system. CSC/SDCE will also establish and operate the MITEC software depot maintenance facility.

   CSC/SDQA is responsible for the independent system evaluation testing of MITEC.

d. Technical Integration Center (TIC), Scott AFB, IL. Two organizations within TIC, DLTS and ETNC, are involved in MITEC.

   TIC/DLTS is responsible for definition of requirements, arbitration of disputes for fielded systems, tracking performance, installed configurations, and lessons learned to determine if MITEC is meeting the baselined requirements. TIC/DLTS is also responsible for advocating modifications and replacements to MITEC.

   TIC/ETNC will develop installation standards for MITEC.

To support software development and planning, testbeds are being constructed, one at 3S, and two at CSC. Hardware for the testbeds is a mixture of communication equipment already in Air Force inventory and equipment purchased specially for MITEC. The latter includes modems, remote line units, test equipment, and the MITEC matrix switch. Procurement of the new equipment has been handled directly by Rome Laboratory.

The software development plan for the Release 1.0 Core CSCI originally called for a Preliminary Design Review (PDR) in April 1991 and a Critical Design Review (CDR) in the July-August time frame. The PDR took place at CSC on 17-18 April, but with the decision to develop MER concurrently with Release 1.0, it was recognized that a CDR at the planned date would not be feasible. Instead, it was decided to hold a second design review at 3S in July to address the new issues introduced by MER. That review took place as planned.

The original development plan also called for a series of six Builds each involving the integration of appropriate modules of the Core and EI CSCIs. The first such Build was scheduled for August. Again, the decision to develop MER caused both the schedule for the builds to be revised and the choice of modules to be changed. The revised schedule calls for Build 1 to be ready for testing at the end of October 1991 and to be made up of the Core and EI modules needed for MER.

In July 1991, the Memorandum of Understanding was rewritten to reflect a 90-day schedule slip due to procurement delays of the matrix switch and the oscilloscope/scanner.

3.1.3 Platform

During the course of the year, a decision was made to change the MITEC Host Computer platform from the AT&T 3B2/600 (3B2) to the Unisys Desktop III. The change was motivated by difficulties encountered with the Verdix Ada Development System (VADS/3B2), the only Ada available for the 3B2. Test programs uncovered a total of 103 bugs in the then available version of VADS/3B2. Six of these were considered to be major risks for MITEC. Additionally, problems were observed with the 3B2 tape software which recovers files from backup tapes.

At the time that the 3B2 was chosen as the host machine, it was the only USAF-standard minicomputer suitable for the task. Subsequently, benchmark tests showed that the Unisys Desktop III, the USAF-standard microcomputer, which uses the popular 386 microprocessor chip, had performance capabilities equal to or greater than the 3B2, and that it should be capable of supporting MITEC except that it lacked the ability to handle a large number of EIA-232 lines. However, by adding a Terminal Server as shown in Figure 3, the required lines could be handled, and the overall configuration would be less expensive than the 3B2 architecture. Further, the MITEC project was already planning to use the Desktop III as the Technical Controller Terminal, and SNL had chosen it for the TCAP system into which MITEC (Early Release) was to be integrated. Switching to that platform would also avoid the need to learn and use two different Ada development systems while working concurrently on Release 1.0 and MER.

A number of the technical reasons to switch the MITEC architecture from the 3B2 to a 386 platform were discussed at the PDR in April. Later, a comparative cost and performance analysis of the two platforms showed that the 386 platform was more cost-effective and powerful than the 3B2. Another factor considered was that the x86 manufacturing community offers a wide variety of commercial products and upward compatibility with future x86 processors. 3S also reviewed other candidate MITEC hosts and concluded that the 386 platform provided the most flexibility for future development at the lowest cost. Capt. Hatten, the MITEC Program Manager, subsequently decided to rehost MITEC (Release 1.0) on a 386 computer, specifically the Desktop III.

Since the Desktop III is in short supply under the Air Force contract, equivalent Northgate 386 machines have been procured for use at CSC/SDCE and 3S for program development use.

3.1.4 Software Development Environment

A major concern when choosing the 386 as the MITEC host platform was identifying a set of off-the-shelf NDCIs which would work together. The following is the set of NDCIs identified as appropriate for the MITEC 386 architecture:

Operating System:	Santa Cruz Operation (SCO) Open Desktop (ODT) Operating System Software (Unix System V)
Ada Compiler:	Alsys Ada Software Development Environment
Database:	Informix OnLine, Informix Structured Query Language (SQL), Informix Fourth Generation Language (4GL), Informix Embedded SQL for Ada (ESQL/Ada), Informix 4GL Rapid Development System (RDS)
Network Interface:	SCO ODT Networking and Communication Software (TCP/IP)

X Windows:	SCO ODT Windowing and Graphic User Interface Software (Motif)

Software development and NDCI evaluation are proceeding with these NDCIs.

The Alsys Ada compiler was selected after some investigation of the Meridian Ada compiler that SNL was using for TCAP. During the investigation 3S discovered that Meridian generated incorrect object code and did not interface fully to Motif. Further, there were doubts as to whether Meridian would interface with Informix, since Informix does not explicitly provide a Meridian interface. Subsequently, 3S passed lessons learned to SNL, and the Alsys Ada compiler was selected for MITEC. 3S has successfully interfaced Alsys 386 to Informix but has encountered a number of problems with the Alsys scheduler. Alsys has been very responsive to our concerns and believes that their recently-shipped Release 5.1 will address the remaining problems.

SCO Open Desktop plus Informix RDBMS have been installed and configured on all 386 development systems. The Alsys compiler is installed on all 386 systems, and Ada Transport Level bindings to UNIX System V network protocols are implemented. A series of tests is currently under way to determine the feasibility of achieving non-blocking performance with multiple Ada tasks in a single Unix process.

An automated front-end to the Source Code Control System (SCCS) has been developed. SCCS is used to aid configuration management of the source code developed under System V Unix. A similar package, developed on the Sun for configuration management of the TRAMCON Event Generator, was used as a model. Additional functionality was added to support the MITEC requirements.

A computer inventory for all hardware, software, and documentation procured for MITEC has been established. A vendor problem log is being maintained. Procedures and forms for checking software in and out of the Configuration Management library are established.

An incompatibility has been identified among the video resolution requirements of MITEC (1024 x 768), the display hardware in the Desktop III (video card and monitor), and the software drivers available with SCO Open Desktop operating system. This problem has been resolved in the short-term by the use of an X terminal as the Technical Controller Terminal for MITEC (Early Release). Options for the longer-term, and MITEC (Release

1.0), are specifying alternate graphics hardware, and/or developing new driver software for SCO UNIX.

The first design for MITEC (Release 1.0) called for the use of the CLIPS expert system shell for operations involving pattern-matching such as (1) finding a duplicate outage and (2) finding two outages that correlate with one another. In order to assess whether or not there were real benefits to be gained from the use of CLIPS, rather than Ada, to implement such operations, these two examples were implemented in both CLIPS and Ada. Comparison showed that although there were some programmer productivity gains from using CLIPS, the limited scope of its intended usage in MITEC would not yield any significant savings. Any such gains from CLIPS would likely be offset by time lost in writing a CLIPS-to-Ada interface. Another problem with CLIPS is the complexity of setting it up to operate in MITEC's multi-threaded environment. Significantly longer run-times were noticed in the CLIPS implementations, an undesirable feature considering the real-time constraints imposed upon MITEC. Finally, there are additional burdens of maintaining CLIPS code and additional memory capacity needed for the CLIPS image. Since these negative issues seem to outweigh the positive gains in productivity and since implementing the required operations in Ada appears to be very manageable, we have decided not to use CLIPS in MITEC (Release 1.0).

3.1.5 Progress and Status

Since MITEC (Release 1.0) development complies with DoD Standard 2167A procedures in which documentation comes before coding, progress in the early stages of the effort is reflected primarily in documents issued and approved. All of the documents described below were issued during FY91 with considerable review and inputs from the other MITEC (Release 1.0) development organizations: CSC/SDCE, RL/C3DA, AFCC, DCEC, and TIC/DLTS.

Two draft versions of the System Design Document (SDD) were issued on the following dates: 13 October 1990 and 30 November 1990. The SDD establishes the system-level design of MITEC (Release 1.0) and is traceable to the MITEC Baseline Specification document (see Appendix A.1 for a copy of the 30 October 1990 version of that document).

The official version of the SDD was issued on 14 January 1991. It represents the Allocated Baseline for MITEC (Release 1.0). Any modifications to the SDD after that date require

Baseline Change Requests (BCRs) that have to be approved by the MITEC Configuration Control Board.

Revision A of the MITEC SDD was issued on 30 September 1991. Revision A incorporates into the MITEC design all of the approved BCRs shown in Appendix A.2 as well as the results of equipment procurement decisions that were not known in January. These include the choice of the Dataswitch Universe/Monolith Plus as the matrix switch and the Tektronix TDS 540 Digital Oscilloscope as the scope to be supported by MITEC.

The Software Development Plan (SDP) for the Core CSCI was reviewed at a 5 November 1990 meeting at 3S. CSC/SDCE pointed out that because some of the devices would not be specified as early in the development process as had been anticipated, the software to handle those devices could not be ready for incorporation into the builds as originally planned. Revision B of the SDP, issued 30 November 1990, reflects a rearrangement of the composition of several software builds previously proposed.

A draft Interface Design Document (IDD) was issued on 14 December 1990. The IDD establishes the design of the interface between the Core CSCI and the EI CSCI. Publication of the next draft of the IDD is deferred until the completion of MER coding.

A Database Technical Operating Report (TOR) was issued 22 March 1991. The TOR reports the findings of a MITEC Database Design Study initiated by LL in October 1989. The TOR describes the use of a database in MITEC, the database requirements and a candidate database design. The results of another study to evaluate candidate commercial database software are included in Appendix A of the TOR.

A preliminary version of the Software User's Manual (SUM) was issued 5 April 1991. The SUM describes how the Technical Controller interacts with the Core CSCI. LL and 3S have extensively reviewed the SUM, and agreed-upon modifications will be incorporated into the next draft after completion of MER coding.

A Draft Core CSCI Test Plan was prepared and distributed at the Design Review II meeting on 19 July 1991.

A number of design issues relative to diagnosis and circuit restoration procedures which have not been explored in the prototype system have been raised and discussed at review meetings. Topics included the use of end-to-end BER tests in fault isolation, trunk vs. circuit restoration decisions, the role of the Circuit Control Office (CCO) in MITEC procedures, and how fault isolation problems should pass from one MITEC installation to another. Final resolution of these and other issues relative to waveform analysis and test point selection that have been addressed in memos has been deferred until MER development is further along.

TIC/ETNC began installation of the MITEC testbed at 3S during July 1991. The racks were bolted to the ground in order to meet earthquake specifications. The FCC-100s were installed and upgraded to Version 4. The FCC-98s and Fireberd 6000 were installed. Installation of the matrix switch is scheduled for November 1991.

3.2 MITEC (Early Release)

3.2.1 System Overview

The principal objective of MITEC (Early Release), also referred to as MER, is to deliver a subset of MITEC (Release 1.0) earlier than the FY93 target date. It is expected that MER will (1) provide Technical Controllers with a desired new circuit test capability, and (2) whet their appetite for the expert system capabilities of MITEC (Release 1.0).

The MER concept involves the integration of two modules of MITEC test instrument code and associated Human Machine Interfaces (HMIs) with the DCEC-sponsored TCAP system being developed by Sandia National Laboratories (SNL). Specifically, MER and TCAP will operate concurrently on the same computer using a common database. Technical Controllers will be able to execute MER and TCAP functions from the same terminal. Integration of TCAP and MER is planned for February 1992 at SNL, and installation at two sites, Ft. Detrick and Andrews AFB, is planned for March-April 1992.

TCAP has two main features: administrative database plus report generation, and remote control of multiple FCC-100 multiplexers. The former stores the station's 1441 card file on line, tracks trouble tickets, and generates various types of reports automatically. The latter accesses the remote control ports of the FCC-100 multiplexers and provides convenient control of the multiplexers from the TCAP screen and keyboard.

MER provides Technical Controllers with computer control of two equipment types: the Hekimian 3701 analog test set and the Fireberd 6000 digital test set. It also allows Technical Controllers to view the layout of any circuit or equipment which has been entered in the TCAP/MER database. MER provides Technical Controllers with a means to enter, recall, and modify test setups for use in controlling the test equipment.

The following is a typical MER scenario involving the convenient execution of circuit tests from the TCAP/MER console:

1. The operator selects a MER icon from a TCAP menu.
2. The operator enters the CCSD of the desired circuit. The configuration of the desired circuit is retrieved from the TCAP/MER database and displayed diagrammatically.

3. The operator selects a testpoint in the circuit to measure and manually patches the test set to the selected testpoint.

4. The operator makes his desired measurements from the TCAP/MER console, using a range of convenient features such as prestored test set configuration files.

5. The operator requests that the test results be printed or saved to disk.

MER functional requirements are defined in a Baseline Specification document which is reproduced in Appendix B of this report.

The MITEC (Release 1.0) development complies with a tailored set of documentation specified in DoD Standard 2167A. MER is intended only as an early delivery of a subset of the MITEC (Release 1.0) system and, as such, complies only with the 2167A documentation standards when MER differs from MITEC (Release 1.0). Three 2167A documents pertaining to MER are planned: the MER Test Plan, the MER Interface Design Document and the MER Software User's Manual.

MER consists of the same two CSCIs found in the Release 1.0 design: Core CSCI and EI. Core is responsible for controlling the basic functions outlined above, the TCAP/MER interaction and the MER Human Machine Interface (HMI). EI is responsible for interfacing with the test equipment. By partitioning the software into these two CSCIs, the detailed equipment characteristics are invisible to Core. Core is being developed by 3S under subcontract to Lincoln Laboratory. EI is being developed by the U.S. Air Force Communications Systems Center (CSC/SDCE).

Figure 4 shows a diagram of the TCAP/MER system. The figure shows one VF and one BER test set, but the design allows for more than one of each. The maximum number that can be handled is not known at this time. The Baseline calls for at least two each. The figure shows FCC-100 multiplexers connected to one port of of the Data Broadcast Unit. The other ports could also connect to FCC-100s, and each port can handle up to ten devices, but communication with TCAP takes place on a one-at-a-time basis since they are all sharing a single port on the Terminal Server.

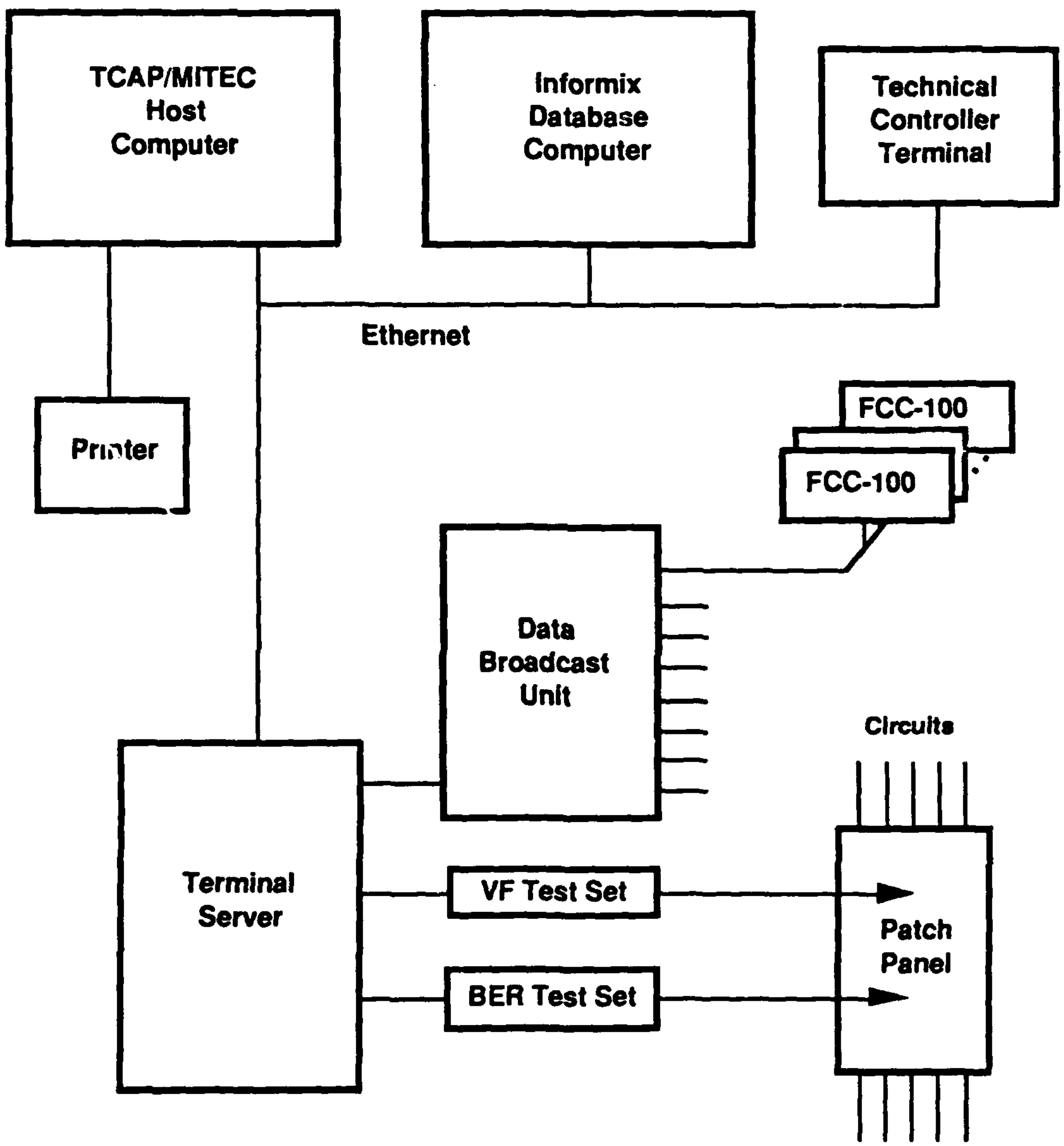


Figure 4. TCAP/MER System Diagram

3.2.2 Progress and Status

Participants from MITEC (Release 1.0) and TCAP jointly developed the specific plans for the TCAP/MER effort in early 1991. DCEC, SNL, 3S, RL, CSC/SDCE, and LL attended two meetings held on 30-31 January 1991 in Washington, D.C., and on 25 February 1991 at Tinker AFB.

On 12 April 1991, CMSgt Gary Drane of the 89th Communications Group at Andrews AFB was briefed on MER test capabilities . During the course of the meeting it was decided that the 89th would become a participant in the MER effort by helping to set system requirements, by providing inputs to the design of the HMI, and by serving as a field test site.

3S and SNL have continually exchanged ideas and results of implementation efforts. Commonalities between the MER and TCAP databases have been identified and documented. A TCAP/MER coordination meeting was held at Sandia National Laboratories (SNL) on 6-7 June 1991. Attendees included LL, 3S and SNL. Issues relating to the 386 platform, schedules, and TCAP/MER HMI integration were discussed and resolved.

A draft Baseline Specification document defining the requirements for MER was approved with minor changes at the Design Review II meeting at 3S in July 1991. Capt. Johnson (AFCC), Capt. Hatten (RL), Mr. Rose (DISA), and CMSgt. Drane (89th Comm Group) will serve as a Configuration Control Board for MER.

CSC/SDCE, 3S, and LL exchanged a large quantity of fax and e-mail correspondence regarding the specific functions to handle the Bit Error Rate and Voice Frequency testing in MER. This effort was focused on the generation of the MER Interface Design Document (IDD) that specifies the messages that will pass between the Core CSCI and the EI CSCI in MER in order to handle the digital and analog testing functions specified by the MER Baseline Specification. The first draft of the IDD was issued on 25 June 1991. A revised draft incorporating all agreed-upon changes for the Bit Error Rate Test functions was issued on 23 August 1991. A further version with agreed-upon changes for VF test functions is in preparation.

The first draft of the MER Test Plan is currently being reviewed.

A prototype of the MER HMI was demonstrated at the MITEC (Release 1.0) Design Review II meeting in July 1991. The MER HMI has since been ported to Alsys and Motif 1.1.

The 89th Communications Group at Andrews AFB was given an overview on 19 August 1991 of the TCAP/MER system and how current TCF procedures will be performed once the system is installed. Technical Controller input was solicited on the design of the MER HMI, test procedures, and logging facilities. The Technical Controllers provided copies of database records and outage reports. 3S has studied the database records and plans to convert some of that data from the dBase III format to the Informix format for incorporation into the TCAP/MER database.

Sections of Core that are on the critical path for Core/EI integration are completed, and the various Core CSCs are integrated with each other. Portions of the Core CSCI that are not on the critical path for integration with the EI CSCI are currently being implemented. Integration will take place with the Build 1 testing that is scheduled for 30 October 1991.

3S and LL attended a TCAP review on 24-25 September 1991 at SNL. SNL plans to ship the TCAP/MER system to Ft. Detrick in March 1992. Unfortunately, Ft. Detrick does not have the VF and BER test sets necessary to exercise MER. When the TCAP portion of the system is operating satisfactorily at Ft. Detrick, the system will be brought up at Andrews AFB. Test sets there will allow the MER sub-system to be used. The overall system is expected to be fully operational in April 1992.

4. Expert Systems for Distributed Control (EDC)

The goal of the Expert Systems for Distributed Control (EDC) program effort is to determine how network control should be distributed throughout the hierarchy of a telecommunications system to maximize adaptation when coping with loss of resources or changes in requirements. To achieve this goal, we are building a simulation environment called NETSIM capable of integrating simulations of three levels of a telecommunications system, the interactions among them, and the effects of control actions taken by expert systems operating at the three levels.

The three simulated levels of communication resources currently integrated into NETSIM are the Transmission level, the DPAS level, and the DSN level as illustrated in Figure 5.

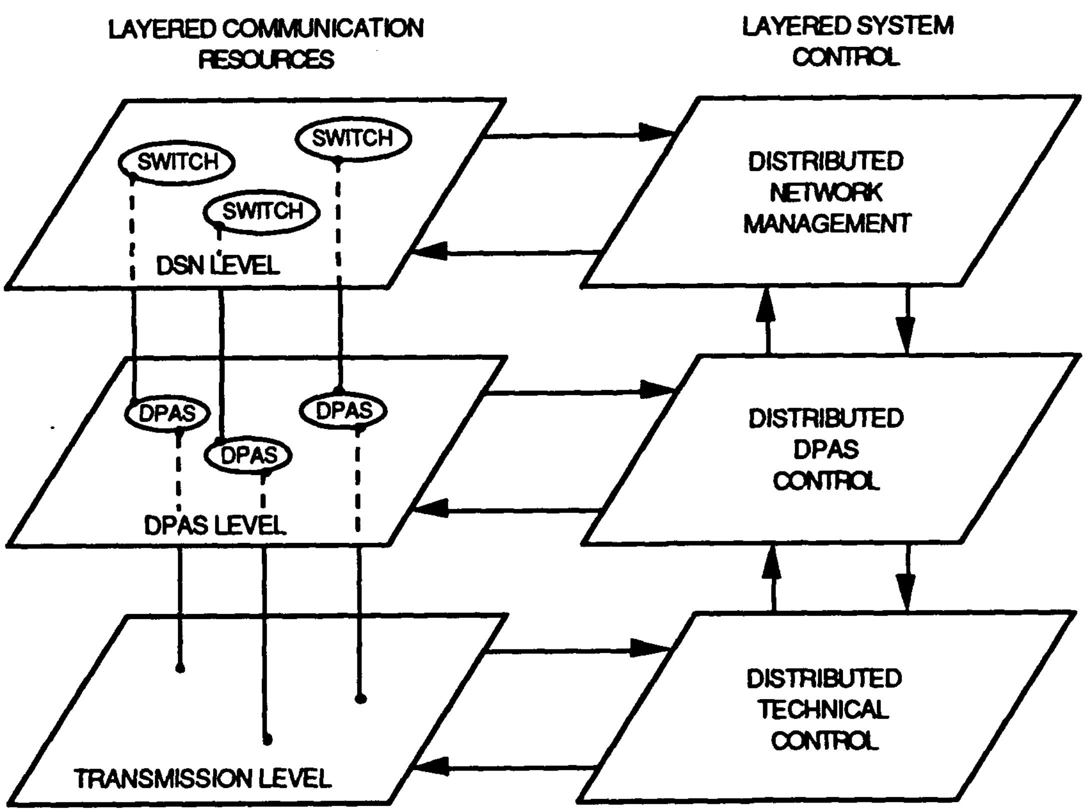


Figure 5. NETSIM layered telecommunications system model

The Transmission level is represented by Digital European Backbone (DEB) segments which consist of analog and digital multiplexers and radio systems. DEB segments are monitored by Transmission Monitoring and Control (TRAMCON) Systems. The TRAMCON Event Generator (TEG) developed in FY90 under DCEC sponsorship is used to simulate faults and their manifestations in the DEB segments.

A network of DPAS nodes linked by T1 trunks represents the DPAS level. NETSIM maintains a representation of these trunks and the circuits they carry. Outages at the Transmission level are automatically reflected as circuit outages at the DSN level.

The DSN level is the switched voice network consisting of end-office switches and tandem switches connected by trunk groups. CCSIM, the call-by-call simulator developed under DCEC sponsorship, provides a simulation of this level. It simulates the behavior of user traffic carried in the network and responds to the effect of circuit outages that are not corrected by control actions at the lower levels. It also provides monitoring of the network's status and allows control actions to be applied at the switch level.

Corresponding to the three levels of communication resources just described, the NETSIM design provides for three levels of expert systems to interact with the communication resources. These are also shown in Figure 5.

At the Transmission level, a distributed network of MITECs would be a natural component of the NETSIM environment. MITECs operate at Technical Control centers in the domain of the Transmission level, detecting lost resources and restoring service. While we have a prototype MITEC in the Laboratory, we do not have a simulated network of MITECs available for integration into NETSIM. Consequently, there is no expert system at the Transmission level currently integrated into NETSIM that can take control actions to deal with problem situations. The TRAMCON Alarm Interpreter (TAI) discussed below, which is integrated with NETSIM, can be used to show how a MITEC would analyze a fault in the transmission system, but it does not provide any restoration functions as a network of MITECs would.

The Autonomous Distributed Routing System (ADRS) algorithm for circuit routing operates in NETSIM with one ADRS process at each DPAS node. The decision to invoke the ADRS algorithm results from the need to find a new route for one or more circuits. Messages to remove a damaged circuit, to find a new path, and to connect the circuit are passed from each node to its neighbors only. In this same manner, messages to find new routes are broadcast throughout the network until either a new path is chosen for a circuit or no more free paths exist. An Expert System operating at the DPAS level, to be developed in the future, would determine when the ADRS algorithm should be run.

The Network Management Expert System (NMES), developed under DCEC sponsorship, detects abnormal traffic conditions at switches or on trunk groups in the DSN and could easily be integrated into NETSIM, but it is a centralized control system and as such it matches only a part of the distributed control goal of the EDC project. Ideally, its functions would be geographically distributed as ADRS is to gain robustness.

Figure 5 shows communications going between the levels of the control hierarchy. We believe that such communication is essential in a hierarchically distributed control system, but it is not yet supported in NETSIM.

In order to distribute control through the different system levels of the network, the specific network representation of a system at one level must be associated with the representation of a system at another level. For example, when a T1 trunk is inoperative at the Transmission level, more than one trunk group at the DSN level may be affected, either totally or partially. This situation in which each individual system views the same network resources differently is an inherent aspect of hierarchically distributed control.

4.1 A Typical NETSIM Demonstration

A typical NETSIM demonstration shows the required simulation capabilities and incorporates several autonomous systems: TEG, ADRS, and CCSIM. A demonstration network consists of DSN switches, DPASs, and a DEBIIA TRAMCON segment.

Since the DSN portion of the network represents the user community, this portion's performance is the center of attention during a demonstration.

The trunk groups providing the links between DSN switches are made up of individual circuits. The groups have names called Common Language Link Identifiers (CLLIs) in CCSIM terminology. The circuits making up the CLLIs are carried individually through the DPASs and/or directly through transmission lines.

To start a demonstration, CCSIM is run to generate normal traffic in the DSN network. NETSIM is then used to interact with TEG in the selection of a simulated fault. A device is chosen in the path of a CLLI whose transmission is provided by the DEBIIA segment. In addition to simulating alarms, TEG also reports failed digroups as a result of the simulated fault. In NETSIM these digroups have a one-to-one relationship to DPAS trunks handled by the ADRS algorithm.

NETSIM determines the affect on the CLLI(s) resulting from the damaged digroups and sends messages to CCSIM indicating the loss of trunk group capacity. As CCSIM continues the DSN simulation, the loss of transmission facilities shows as degraded performance for the DSN users.

The demonstration then continues by using NETSIM to invoke ADRS with the request to find new routes through the DPAS layer to restore service for the affected CLLI(s). Depending on the resources available, ADRS may find new routes through the DPAS network for all or only some of the damaged circuits. NETSIM displays the ADRS results and sends a simulation message to CCSIM indicating restoral. CCSIM then shows the extent of improvement in DSN service that is obtained.

4.2 Building NETSIM Networks

NETSIM is usually invoked with the name of a predefined network. However, the user may optionally enter a name of a nonexistent network configuration. In that case, NETSIM displays a blank screen where the user can graphically construct a network. Network construction is menu driven, and begins at the highest level in the circuit hierarchy, the network level. Here the user specifies the sites and connections between the sites in the network. At this level, the user is not concerned with details of either the sites or the connections; rather, the user is simply specifying which sites are connected to which other sites. Once the network connectivity is defined, the user invokes the site editor which permits the user to enter the equipment and the connectivity of all the equipment at each site. This editor is also menu driven. Some domain knowledge is maintained in the site editor so that if an illegal configuration of equipment is proposed, the user is notified. For example, the site editor issues a warning if the user tries to connect equipment ports with mismatching bit rates.

4.3 NETSIM Integration with TAI

During FY91 an expert system called the TRAMCON Alarm Interpreter (TAI) was developed under DCEC sponsorship. Its function is to interpret the constellations of alarms that result from faults in the pieces of transmission equipment monitored by TRAMCON and to provide lists of possible faults that might have generated the alarms. In general, the relationship between faults and alarms is not unique, i.e., there is more than one fault that can cause any particular constellation of alarms to be reported. Since we had observed that NETSIM's graphics capabilities provided a convenient mechanism for entering faults for the TRAMCON Event Generator (TEG), we decided to use NETSIM as a vehicle for running and demonstrating TAI as well, and we extended NETSIM appropriately.

In operation with TAI, NETSIM provides a mechanism for the user to enter the name of a file that contains a list of alarms and to initiate TAI. Upon TAI's completion, NETSIM indicates which devices may have caused the alarms by changing the color of the specific devices and their sites. The file of alarms that is the input to TAI is normally obtained by running NETSIM in the TEG mode and saving the alarms to a file.

During the last quarter of FY91, TEG and TAI with NETSIM as their user interface, were demonstrated to representatives of DISA/DCEC/DRTV, AFSC/RL/C3DA, and AFCC/CCS plus a senior TRAMCON system operator from USAFE. The potential utility to TRAMCON system operators of a system like TAI was apparent to all observers. Significant interest has been expressed in deploying a TAI-like adjunct to TRAMCON systems in a few years.

4.4 NETSIM Delivery

In June 1991 a TEG/NETSIM integrated system was installed and demonstrated at DCEC. Additionally, instruction was provided on the mechanics of performing the demonstration.

Appendix A. MITEC (Release 1.0) Baseline Specification

A.1 The Baseline Specification as of 30 October 1990

1. Physical requirements

    a. Memory
        1. Development systems - 64MB: Required to support 4-6 ADA developers (Two systems at CCSC, one system at 3S)
        2. Operational system - 16MB (projected)
    b. Disk Storage
        1. 1 300MB hard disk
        2. 1 300MB removable hard disk
    c. Backup storage - 1 125MB tape drive
    d. Printers
        1. one Laser printer: 8 page/min. (RS 232 interface, honor XON and XOFF)
        2. one 132 column printer (RS 232 interface, honor XON and XOFF)
    e. ADA compiler - Verdix
    f. Operating system - Unix V, Version 3.2.2
    g. I/O
        1. 2 FXM cards (128 ports) - Operational systems
        2. 1 FXM card (64 ports) - Development systems
    h. Graphics Terminals
        1. Number; 2 - 4 depending on installation size
        2. Type: Desk Top III 20", VGA (1024X768)
        3. Interface to 3B2
            a. Ethernet
            b. X-windows
    i. Device protocols:
        1. RS-232
        2. IEEE 488 (obtain through 3rd party vendor)

2. Communications Equipment interfaces

    a. FCC-100 (versions 1&2); minimum of 50 devices
    b. FCC-98; minimum of 128 devices.
    c. DPAS; Release 1.0 alarm inputs only; Release 2.0 full monitor and patching capability; minimum of 160 digroups; Snyder protocol
    d. Remote Line Units
        1. Analog
        2. Wrap-around
    e. Datalok 10E

3. Circuit Test and Access/Patching Equipment interfaces

    a. Matrix Switch - TBD
        1. Interfaces required
            a. RS-232 - 40%
            b. RS-449 - ref MIL STD 188-114
            c. MIL STD 188-114 - 10%
            d. V.35 - 10%

e. VF, data grade lines, 300 to 3300 Hz - 10%
2. Number of monitor ports
a. 400 circuit installation
1. 1 VF (HLI 3701, IEEE 488.1 interface)
2. 2 digital (one ea Fireberd and O'scope)
b. 800 circuit installation
1. 2 VF (2 ea HLI 3701)
2. 4 digital (2 ea Fireberd and O'scope)
b. Access Switch - HLI 3200
a. 400 circuit installation
1. 1 VF (HLI 3701, IEEE 488.1 interface)
2. 1 digital (Fireberd)
b. 800 circuit installation
1. 2 VF (2 ea HLI 3701, IEEE 488.1 interface)
2. 2 digital (2 ea Fireberd)
c. VF testset - HLI 3701
d. BERT - Fireberd 6000
e. Digital storage scope or A/D - TBD
f. RLU
1. Analog
2. Wrap-around
g. FCC 100 (version I & II) polling: use port provided by FXM card
h. Datalok 10E
i. Perform device initializations from crashes
1. Pacify - device to known state
2. Reset - match MITEC database
3. Dump - input device database or status

4. Circuits

a. Digital (Monitor alarms for 1.544 and 2.048 MBS, fault isolate and restore circuits up to 128 KBS)
b. 4 KHz Voice Frequency
c. Configuration
1. Point-to-point
a. Full duplex
b. Half duplex
2. Broadcast - one way

5. Database

a. Informix (4GL)
b. Question/response for circuits data entry with error checking where possible.
c. Process to detect errors between computer's database and device's database with user notification to resolve conflicts.
d. Device availability
1. broken/fixed
2. currently in use for higher precedence circuit
e. circuit information
1. CCSD, primary key
2. Precedence
a. Priority Schemes Supported:
1. RP
2. TSP

b. MECL
c. manual override to freeze asset
3. Multiple CCSDs for 1 circuit, will use alias to maintain primary key.
4. Possible circuit states
a. normal
b. actual
c. planned; circuit may be obligated against up to 20 restoral plans
d. preempted
5. Restoral plans

6. Diagnosis

a. Types of problems addressed (project resolving approximately 70% of daily outages).
   1. Signal, no signal
   2. Incorrect signal
b. Override for troubleshooting modes, in order of priority (assumes all circuits have equal restoral precedence).
c. Method to reliably hand off problem to:
   1. peer MITEC
   2. operator (problem not resolvable by MITEC)
   3. desirable: I/O with adjacent unmanned TCFs, to allow unmanning of midshifts.
d. MITEC to MITEC will cooperatively fault isolate and restore failures by exchanging local information.
e. MITEC will operate in three troubleshooting modes:
   1. Automatic - MITEC will perform *all* aspects of troubleshooting with no operate input required.
   2. Lockstep - MITEC will determine strategy for troubleshooting, but operator must confirm each step.
   3. Manual - MITEC will be directed by operator for all aspects of troubleshooting.

7. Patching

a. Only local MITEC will implement patch in local station, adjacent MITECs will not be able to implement distant end patch. Patches should be made in <30 sec from request.
b. Patch on top of patch is not represented, only normal circuit path and current patch configuration.
c. Restoration patches, resulting from fault isolation, are only made between adjacent MITECs, no three node restorations.
d. MITEC is smart, only reasonable patches will be allowed, i.e., VF signal applied to digital input.
e. MITEC will not allow patches which causes assets to be isolated from computer control requiring operator intervention to "undo".
f. MITEC will coordinate patches with peers and notify operator when adjacent TCF is manual.
g. All patching will be accomplished via matrix switch in release 1.0. (No manual patching is allowed or required for circuits controlled by MITEC).
h. MITEC will perform alt-routes using standard patching operations and conventions IAW DCAC 310-70-1.
i. MITEC will implement preempts using standard preempting operations and conventions IAW DCAC 310-70-1.

8. Quality Control

   a. Release 1.0 will not perform in-service, out-of-service, and PMPs.
   b. MITECs will be designed to accommodate this feature in future releases.

9. Alarms

   a. devices monitored for alarms
      1. FCC 100
      2. FCC 98
      3. DPAS
   b. Deductive knowledge of (for alarm/diagnostic purposes; Direct interface to these devices will be P3I in future releases):
      1. FCC-99
      2. Radios
   c. Types of alarms:
      1. hard alarms, continuous
         a. MITEC will automatically fault isolate
         b. MITEC will automatically restore service when possible
      2. intermittent alarms
   d. will perform alarm filtering
   e. will automatically clear secondary alarms when primary is addressed

10. TCF-TCF Communication

   a. Primary, two options
      1. dedicated orderwire between adjacent stations using service channel, max number 8 with capability to be expanded to 12.
      2. DDN orderwire connection
   b. Backup circuit testing: dial-up
   c. MITEC TO NON-MITEC: operator notified and must proceed in lockstep or manual fault isolation mode (dumb terminal at manual node desirable.
   d. acknowledgment strategy for MITEC to MITEC
      1. honor busy notification, working on XXXX problem.
      2. Terminate previously requested process.

11. System Load, MITEC release 1.0

   a. max number of circuits in database, 4000.
   b. max number of adjacent nodes required to interface to 8, with the capability to be expanded to 12.
   c. Number of concurrent diagnostic problems addressed will be a minimum of four; circuit actions exceeding four will listed in a buffer in precedence order.

12. Security Mechanisms

   a. Login with password and only audit trails thereafter.
   b. Dial-up/DDN access protection will be through digital signature.
   c. Recognize when excessive failed attempts to provide correct digital signature.
   d. capability to change digital signature.

13. Reports:

a. MITEC Release 1.0 will not interface to CATC or perform admin functions.
b. MITEC will print hard copy of information on circuits and fault isolation tests.
c. MITEC database will contain information which will permit admin functions in future releases.

14. Tailored 2167A documentation.

15. Help facilities for:

a. data entry.
b. All other areas will not require help, designed to be self explanatory
c. MITEC is smart and will only allow reasonable actions to be accomplished.

16. Training Facilities: not necessary, MITEC is not a training tool.

If personnel wish to conduct training, spare assets maybe used to enter dummy circuits in database and training accomplished on spare assets.

17. MITEC will not contain, process classified information or operate in a red environment.

18. Human Machine Interface guidelines

a. Mouse will land on default positions whenever appropriate.
b. Everything done with mouse can be done with keyboard.
c. Mouse windows will be large for easy, rapid point and shoot.
d. Colors will maintain consistent representation of information.
e. Like input/functions will always be in same area of VDU.

19. Recovery events

a. system crash.
b. power outages/flux
c. failed device fixed
d. test equipment fixed
e. loss of communications with peer MITEC

A.2 Baseline Change Requests

Thirteen Baseline Change Requests (BCRs) were prepared and submitted to CSC/SDCE during FY91.

The MITEC (Release 1.0) Configuration Control Board has approved the following twelve BCRs:

1. Interface to FCC-100 (V) 4 and 4X instead of FCC-100 (V) 1 multiplexers.

2. Eliminate requirement to infer higher-level outages from DACS alarms.

3. Use SCO Motif 1.1 to interface between the Technical Controller terminal and the CORE CSCI.

4. Eliminate requirement for a maximum number of restoral plans against which a circuit may be obligated.

5. Use an Ethernet Terminal Server interface between the MITEC host computer and test/access equipment.

6. Allow EI CSCI to communicate via Ethernet or EIA-232.

7. Receive DACS alarms from Datalok rather than DCT Admin Port.

8. Eliminate requirement for backup remote communications.

9. Use a router for DDN remote communications.

10. Change MITEC host computer from 3B2 to 386.

11. Change login/logoff procedures.

12. Adopt color-blind conventions.

A thirteenth BCR to include Hekimian Labs 3901 M-RA as an additional VF test set has been provisionally approved. The Hekimian 3901 M-RA will be incorporated if the interface code is transparent to the Core CSCI.

Appendix B. MITEC (Early Release) Baseline Specification

B.1 The MER Baseline Document as of 30 September 1991

1. Goals:

   a. Will be an early release of MITEC Release 1 manual mode capabilities.
   b. Will execute concurrently with TCAP and allow the operator to switch between MITEC and TCAP.
   c. Will support both acceptance testing and troubleshooting modes of operating the test instruments. The acceptance testing mode supports the tech controller not attending a test, thus requiring test results to be logged over time. The troubleshooting mode supports the tech controller when rapid interaction with the device is required.
   d. Will use front-panel terminology of the test instruments, where applicable, in parameter acquisition and result presentation.

2. Physical Requirements:

   a. Will interface to at least two each of the vf and digital test instruments, without excluding the option to increase the number of either test instrument.

3. Digital Testing Capabilities:

   a. Will interface to the TTC Fireberd 6000 with the RS-232 interface.
   b. Will allow for eventual control of all Fireberd 6000 features, but the initial implementation will exclude:
      1. Jitter measurements
      2. Histogram analysis
      3. Customization of results
   c. Will support the following interfaces:
      1. Internal RS-232
      2. V.35 (Model 40202)
      3. RS-449 (Model 40200)
      4. DS1/T1 (Model 40540) - TBD subset
   d. Will allow the operator to interrupt BER tests at any time.
   e. Will provide feedback and logging comparable to the Fireberd 6000 capabilities, as the test progresses. Feedback will be provided at a frequency of up to 12 times per minute. For logging, a maximum rate of TBD is acceptable.

4. VF Testing Capabilities:

   a. Will interface to the Hekimian 3701 test set.
   b. Will be capable of controlling a remote VF test set in a master/slave setup, using the line under test.
   c. Will provide the following test capabilities:
      1. Test Tone Level
      2. Frequency Response
      3. Envelope Delay
      4. Signal to C-Notched Noise Ratio
      5. Maximum Net Loss Variation

6. Max Change in Audio Frequency
7. C-Message Noise
8. C-Notched Noise
9. Impulse Noise
10. Phase Jitter
11. Peak to Average Ratio
12. Nonlinear Distortion
13. Terminal Impedance

5. Test Results:

   a. Will maintain a test log which includes: the date, time, operator initials, circuit information (optional), test parameters and the requested test results.
   b. Will provide the operator a way to view both the instantaneous test results and the test log during a test.
   c. Will provide the option to hardcopy both the test results and the test log.

6. Relationship to Database:

   a. Will allow test instrument control independent of whether or not circuits are in the database.
   b. Will graphically depict the circuit layout if the circuit path information is in the database.
   c. Will provide defaults for test setup parameters based on the circuit information that is in the database.

Glossary

ADRS	Autonomous Distributed Routing System, the algorithm used at the DPAS level in NETSIM to find routes for circuits
AFCC	Air Force Communications Command
AFNET	Air Force Integrated Telecommunications Network
AFSC	Air Force Systems Command
BCR	Baseline Change Request
BER	Bit Error Rate
BERT	Bit Error Rate Tester
CCSD	Command Communications Service Designator, the name used to identify circuits in MITEC and in TCFs
CCSIM	the Lincoln Call-by-Call Simulator developed under DCEC sponsorship
CDR	Critical Design Review
CLIPS	'C' Language Integrated Production System, an expert system shell developed and supported by NASA
Core	the CSCI of the production MITECs being developed by LL and 3S
CSC	the Air Force Communications Systems Center, Tinker AFB, OK
CSCI	Computer Software Configuration Item
DACS	Digital Access and Cross-connect System, the ATT hardware used in DPAS
Datalok	the alarm sensing device used in MITEC
DCA	Defense Communications Agency, old name for DISA
DCEC	Defense Communications Engineering Center, Reston, VA
DEB	Digital European Backbone, part of the U.S. microwave system deployed in Europe
DISA	Defense Information Systems Agency, formerly called DCA
DPAS	Digital Patch and Access System
DSN	Defense Switched Network
EDC	Expert Systems for Distributed Control, a LL research project
EI	Equipment Interface, the CSCI of the production MITECs being developed by CSC/SDCE
FCC-98	the military standard T1-rate multiplexer (AN/FCC-98) supported by MITEC

FCC-100	the military standard low-speed time-division multiplexer (AN/FCC-100) supported by MITEC and TCAP
Fireberd 6000	the bit error rate tester (BERT) supported by MITEC
HMI	Human-Machine Interface
IDD	Interface Design Document
IDNX	Integrated Digital Network Exchange, the switch chosen for AFNET
LL	M.I.T. Lincoln Laboratory
LSTDM	Low Speed Time Division Multiplexer
MER	MITEC (Early Release)
NDCI	Non-Developmental Configuration Item
NETSIM	the simulator developed to support the EDC research activity
PDR	Preliminary Design Review
RADC	Rome Air Development Center, old name for Rome Laboratory
RL	Rome Laboratory, Griffiss AFB, NY, formerly called RADC
SDD	System Design Document
SNL	Sandia National Laboratories
TAI	TRAMCON Alarm Interpreter
TCAP	Technical Control Automation Project
TCF	Technical Control Facility
TEG	TRAMCON Event Generator, a TRAMCON simulator developed by LL and 3S under DCEC sponsorship
TIC	Technical Integration Center, Scott AFB, IL
TRAMCON	the Transmission Monitoring and Control system used with the DEB microwave system
VF	Voice Frequency
3S	Structured Systems and Software, Inc., Laguna Hills, CA

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (*Leave blank*)

2. REPORT DATE
30 September 1991

3. REPORT TYPE AND DATES COVERED
Annual Report, 1 October 1990 – 30 September 1991

4. TITLE AND SUBTITLE
Knowledge-Based System Analysis and Control

5. FUNDING NUMBERS
C — F19628-90-C-0002
PE — 33126F
PR — 295

6. AUTHOR(S)
Harold M. Heggestad

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Lincoln Laboratory, MIT
P.O. Box 73
Lexington, MA 02173-9108

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
RL/C3D4
Griffiss AFB, NY 13441

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
ESD-TR-91-243

11. SUPPLEMENTARY NOTES
None

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (*Maximum 200 words*)

A major aspect of work in FY91 has been the development of MITEC systems for field deployment. A decision was made by Rome Laboratory and the Air Force Communication Command to undertake the development of an Early Release version of MITEC for delivery in FY92 as well as the MITEC (Release 1.0) system started in FY90 and scheduled for delivery in early FY93. The development plan involves the cooperative effort of three Air Force organizations: Rome Laboratory (RL/C3DA), the Air Force Communications Command Communications Systems Center (AFCC/CSC), and Technical Integration Center (AFCC/TIC), with Lincoln Laboratory and its subcontractor Structured Systems and Software, Inc. (3S). The decision to develop MITEC (Early Release), also referred to as MER, in parallel with MITEC (Release 1.0) has had a major impact on the project. The design has been changed, a different host computer has been chosen, and schedules have been altered to make the portions needed for MER available earlier.

Work on the prototype MITEC systems in operation at Lincoln Laboratory and at Rome Laboratory has continued with the effort focused on providing input to the design of the production MITEC systems. A bit error rate (BER) testing capability was added, waveform analysis and presentation were enhanced, and alarm polling and logging capabilities were extended. Software infrastructure improvements were carried out to support these new features, and the browse subsystem was enhanced. Changes were made in the testbed equipment at Lincoln Laboratory, and site visits were made and documentation generated in support of the testbeds at Rome Laboratory. Experiments explored alarm correlations, the effect of T1 jitter on equipment in the testbed, and the master/slave capabilities of the VF test sets.

Work was also carried out in the Expert Systems for Distributed Control (EDC) research area. The goal of this research is to determine how network control should be distributed throughout the hierarchy of a telecommunications system to maximize adaptation when coping with loss of resources or changes in requirements. The NETSIM simulation environment built to achieve this goal can now integrate simulations of the Transmission, Digital Patch and Access System (DPAS), and Defense Switched Network (DSN) levels of the communication system. A typical demonstration shows the effect of damage at the Transmission level on performance as seen by DSN users as well as the ability of the Autonomous Distributed Routing System (ADRS) algorithm to compensate for trunk outages by changing circuit routing at the DPAS level. NETSIM is also now integrated with the TRAMCON Alarm Interpreter (TAI) developed under DCEC sponsorship.

14. SUBJECT TERMS

expert systems	telecommunications
tech control	machine intelligence
tech control automation	Defense Communications Systems

15. NUMBER OF PAGES
60

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	SAR